EUROGRAPHICS Symposium on Sketch-Based Interfaces and Modeling (2011)
T. Hammond and A. Nealen (Editors)

Now or Later: An Initial Exploration into User Perception of Mathematical Expression Recognition Feedback

Jared N. Bott, Daniel Gabriele and Joseph J. LaViola Jr.

University of Central Florida, Department of EECS, Orlando, FL, USA

Abstract

Mathematical handwriting recognition is an important method of mathematics input for computers. While strides in recognition have been made in recent years, recognition is still at a level where mistakes are common and often inexplicable from the user's point-of-view. As a result, recognition mistakes can cause user distraction and frustration. We examine how user preference for real-time or batch recognition mode is affected by recognition accuracy and the number of expressions entered. Our results show that users prefer real-time recognition when working with multiple expressions; however, with high accuracy recognition, users did not prefer one recognition mode over the other.

Categories and Subject Descriptors (according to ACM CCS): Information Interfaces and Presentation [H.5.2]: User Interfaces—Interaction Styles; Mathematics of Computing [G.4]: Mathematical Software—User Interfaces

1. Introduction

Mathematical handwriting recognizers are complicated, near-inscrutable black boxes from a user's point of view. A small change in the way one writes a symbol can have a large effect on recognized expressions. While mathematical handwriting recognizers are mostly generalized in terms of the mathematics they support, they can also show a narrow focus in their preference for one result over another. We cannot expect that mathematical handwriting recognition will achieve perfect accuracy in the short term, perhaps not even in the long term. Not only is there great variation in handwriting, but mathematics add a 2D layout component to the problem. Consequently, we need to examine how best to allow users to focus on entering mathematics and not on the recognizer's inaccuracies.

Invoking handwriting recognition comes in two flavors, now (real-time recognition) and later (batch recognition). Real-time handwriting recognition can be a distraction for some users, causing them to slow down to fix mistakes; conversely, it can provide valuable feedback by allowing a user to see what went wrong and when. Batch recognition, where the user writes an entire expression (or group of them) and then manually invokes recognition, can allow a user to write without distraction, but provides no recognition feedback during writing, often requiring the user to perform some action to invoke recognition. Batch recognition also requires visual processing and subsequent correction of a potentially large amount of errors at once. The parameters in mathematics recognizers greatly affect their usability and their users' feelings.

To explore these issues, we performed a study to see how users felt about these recognition modes. In order to perform the study we used a Wizard of Oz system for simulating mathematics handwriting recognition systems, the WOZ Math Recognizer (see Figure 1) [BL11]. Twenty-four users participated in our experiment with both recognition modes, three recognition accuracy levels, and different expression set sizes.

2. Related Work

While mathematical handwriting recognition has been explored in detail, most research focuses upon recognition algorithms. How to measure the accuracy of a recognized handwritten equation is the subject of much debate [CY01,NP95,SNA99]. For instance, Anthony used Levenshtein word distance as the measurement for symbol accuracy [Ant08]. LaViola used two metrics, a symbol accuracy metric and a position accuracy metric in [LaV07]. We have

SBIM 2011, Vancouver, British Columbia, Canada, August 5 – 7, 2011.
© 2011 ACM 978-1-4503-0906-6/11/0008 $10.00

taken a similar approach, using both a symbol and a position accuracy metric.

The effects of different accuracy rates on users are rarely explored; this is not surprising as handwriting recognizers are not designed to have controlled accuracy, making it hard to perform such studies. Martin et al. examined the effect of task on recognition accuracy in [FGP*09] and looked at how gathering training data using three different tasks (isolated shape copying, diagram copying, and diagram synthesis) affected classification. They found that training from different tasks' data had little effect on accuracy when using a "factory-trained" recognizer, one which has not been specifically trained for the person using it.

Recognition feedback is another area that has been less thoroughly explored. In [LLMZ08], LaViola et al. explore different methods for displaying recognized mathematical expressions and conclude that small, typeset expressions displayed below the user's writing was most preferred. However, they did not control for recognition accuracy or explore recognition feedback modes. Wais et al. explored a variety of methods for triggering recognition, indicating strokes to recognize, displaying recognition feedback, and the effects of common recognition errors on user experience [WWA07]. Their focus was upon sketch recognition and did not explore recognition accuracy, nor mathematical handwriting recognition.

Previous research has been performed using Wizard of Oz studies in relation to handwriting recognition. LaLomia performed a study to determine acceptable handwriting recognition rates for basic writing tasks that did not involve math [LaL94]. The study used randomly introduced errors to reach the target accuracy rate and considered the result's acceptability for a variety of tasks. Read et al. performed a similar experiment with children using batch recognition [RMC03]. Neither study focused on mathematics. We think that these results are not applicable to the math domain because math has a complex structure not found in normal writing. Oviatt, et al. use a Wizard of Oz scenario to examine how students adapt their computer input while using a dual input (speech and pen) system while solving mathematical problems [OSA08]. Specifically, they looked at how the students changed their vocal levels and pen pressure when the computer system did not recognize it was being communicated with. Anthony's work on mathematics input for intelligent tutoring explores how different modes of input affect user learning, input speed, and cognitive load among other things, and included the use of Wizard of Oz studies in some cases [Ant08].

3. Experimental Study

In order to explore whether people preferred real-time or batch recognition for a mathematics recognition task, we conducted a user study. Our primary concern in performing the study was determining participant preference for recognition mode. Prior to the experiment, we formulated several hypotheses:

- As recognition accuracy increases, user preference for real-time recognition will increase. Participants will prefer immediate feedback when it is mostly correct.
- Accuracy will affect user feelings. Participants will report less frustration and distraction, and more ease in writing and correcting when working with more accurate recognizers.
- Equation set size will affect user feelings. Participants will report increased distraction and frustration with the larger set size. The increased number of errors will be more to find and fix at a time, making it more frustrating and distracting.

3.1. Subjects and Apparatus

We recruited 24 college students from the general university population (18 male, 6 female) to participate in our study. The participants ages ranged from 18 to 31. Twelve participants had previous experience using tablet PCs, while eight had used some form of handwriting recognition software, and three had used mathematical handwriting recognition software. We had one left-handed participant. The experiment took approximately 1.5 to 2 hours to complete and each participant was paid 10 dollars for their time.

Our experimental setup consisted of two workstations, the participant station and the wizard station. The participant station was an HP Compaq tc4400, 12.1 inch tablet PC running Windows XP Tablet Edition. The participant station was cordoned off from the wizard station in order to remove distractions for the participant and minimize any noises from the wizard. The sound of a fan was also played during the experiment to further minimize the sounds. For the wizard station, a 21 inch monitor displayed the wizard interface for the WOZ Math Recognizer [BL11], and a secondary 17 inch monitor showed the participant's screen (Figure 1a). A desktop PC with two Intel Core i7 920 processors at 2.67 GHz and 9 GB memory running Windows 7 powered the wizard station. Two people were required to administer the experiment, a proctor and a wizard.

3.2. Experimental Task

Participants were asked to complete six writing tasks; each consisted of writing expressions varied in size, at one of three recognition accuracy levels, which apply to both symbol and position accuracy, and in one of two recognition modes. Expression accuracy is measured in two ways, symbol accuracy and position accuracy. Symbol accuracy is measured as the number of correct symbols divided by the total number of symbols, and position accuracy is measured by dividing the correct number of parsing decisions by the total number of parsing decisions (see [BL11] for details).

Figure 1: The wizard station (a) is composed of a desktop PC and two monitors, a 21" monitor (center) and a 17" monitor (left). The recognition station (b) is a tablet PC.

For each accuracy level, we used the same accuracy targets for each measure, symbol and position. The three accuracy levels (90%, 95%, and 99%) were chosen because we felt they were reasonable accuracy levels; any lower and users would likely find them too hard to use. That is to say that 90% accuracy is the minimum accuracy that we believe that users might find tolerable. Two tasks were performed at each accuracy level; one task used a set of single expressions (single equation set) consisting of five expressions written individually, and the other task had a set of multiple expressions (multiple equation set) consisting of five groups of three expressions. For each task, participants performed two subtasks, each time writing the same expression set, once in batch recognition mode, once in real-time recognition mode. Overall, each participant performed six tasks, writing 120 expressions in total (60 unique expressions, each written twice).

Each task has its own set of expressions, so we constructed six equation sets. Single equation set tasks had five separate expression and multiple equation set tasks had five groups of three equations. For the experimental tasks, we designed our expression sets to be in all lowercase; capitalization errors were also disabled. In real recognizers we examined [Mic09, ZMLL08], changing case through erasing and rewriting was problematic at best; recognizers tend to solve this problem by providing functionality to allow the user to choose from a list of alternate recognized expressions. Consequently, we chose to avoid this issue altogether.

3.3. Experimental Design and Procedure

We used a 3 by 2 by 2 within-subjects factorial design, where the independent variables were recognition accuracy, recognition mode, and set size. Two recognition modes, batch recognition and real-time recognition, were included; two set sizes were used, one equation by itself and three equations together. We felt that three equations as a group were a good compromise in terms of time spent writing and group size. The dependent variables were user preference for recognition mode and distraction level, which were determined through a questionnaire given after each recognition task.

A proctor guided the participants throughout the experiment, giving them questionnaires, finding mistakes in what they wrote, and performing interviews. First, the participants were given a pre-questionnaire. The pre-questionnaire asked the participants for their age, gender, which hand they write with, as well as whether they had ever used a tablet PC, handwriting recognition, or a mathematics recognizer. They then practiced using the different recognition modes. Participants were then given a preliminary task to familiarize themselves with the recognizer interface. During the explanation of the study and the interface, participants were told that they would experience different recognition accuracies during the experiment. While working with batch recognition mode, participants wrote and corrected two expressions, and then proceeded to write and correct a multiple equation group of three expressions in real-time recognition mode. Participants were then given a series of tasks to perform. The order in which participants worked through the different tasks was randomized and counterbalance such that one-third of the participants received the 90% accuracy tasks first, one-third received the 95% accuracy tasks first, and one-third received the 99% accuracy tasks first. The presentation of the multiple equation set task or the single equation set task first was also counterbalanced. As each task has two subtasks, one in real-time recognition mode and one in batch recognition mode, three of the tasks were performed with the real-time recognition subtask first, and three of the tasks were performed with batch recognition first. The participants were instructed to find their mistakes and correct them before moving on to the next expression; the proctor pointed out errors that they missed when necessary.

The post-task questionnaire asked subjects to rate their agreement with four statements on a seven point Likert scale, where 1 was Strongly Disagree, 4 was Neutral, and 7 was Strongly Agree:

- Easy to write: It was easy to write the expressions.
- Easy to correct: It was easy to correct the expressions when necessary.
- Frustration: It was frustrating writing and correcting the expressions when necessary.
- Distraction: I was distracted from writing expressions by the recognition system.

An interview after each task pair was also given, which asked which recognition mode the participant preferred for the previous task pair. In a final interview after all tasks were completed, we asked participants a few brief questions about

Table 1: Position and symbol totals for single and multiple equation sets. We balanced the sets so that they had similar position and symbol counts.

Multiple Expressions	Set 1	Set 2	Set 3
Positions	310	321	317
Symbols	223	205	204
Single Expression	**Set 1**	**Set 2**	**Set 3**
Positions	220	213	214
Symbols	98	102	99

$$3x^3 + x^2 + 1 = 8 \quad (1)$$

$$\int\int xy^2 + x^2 y \, dy \, dx \quad (2)$$

$$p(t) = \cos(t - e) + \sin(t + k) \quad (3)$$

Figure 2: Example expressions used in our experiment.

the two recognition modes, such as whether they changed the way they wrote during the experiment, whether they watched the real-time recognized expressions as they wrote, and what they thought about how the recognized expressions were displayed.

We designed our expression sets so that they had similar numbers of symbols and positions (see Table 1). Most expressions were basic polynomial equations. Within the multiple equation sets, there were trigonometric equations in each set. The single equation sets all had at least one trigonometric equation and one integral. Example expressions can be seen in Figure 2.

3.4. Results

We examined user's preferences for batch or real-time recognition for each expression set size at each accuracy level using chi-square tests (see Table 2). Most participants did not exclusively prefer one recognition mode to the exclusion of the other. Eighteen participants preferred batch for at least one of the six tasks and twenty-three participants preferred real-time for at least one of the six tasks. For multiple equation sets, there was a clear preference for real-time recognition at all three accuracy levels ($\chi_1^2 = 10.67$, $p < 0.05$). For single equation sets, at 90% accuracy there was also a preference for real-time recognition ($\chi_1^2 = 5$, $p < 0.05$), but there was no clear preference at higher accuracy levels. Using contingency tables, we examined participant preference for recognition mode. Looking at accuracy, there was no significance in preference across the three accuracy levels ($\chi_1^2 = 1.48$, $p = 0.48$). When we looked at equation set sizes, there was statistical significance ($\chi_1^2 = 7.91$, $p < 0.005$), meaning that there was a difference in preference for recognition mode between the single and multiple equation set tasks.

Table 2: User preference statistics for batch and real-time recognition for each accuracy level and task set size. In most cases, there was a statistical preference for real-time recognition.

		Batch	Real-time	χ^2	p
90%	Single	6	18	6	$p < 0.05$
	Multiple	4	20	10.67	$p < 0.01$
95%	Single	11	13	0.167	$p = 0.683$
	Multiple	4	20	10.67	$p < 0.01$
99%	Single	10	14	0.667	$p = 0.414$
	Multiple	4	20	10.67	$p < 0.01$

Table 3: Mean perceived recognition accuracies. Participants showed a clear underestimation of accuracies and had a greater underestimation for the multiple expression set tasks.

Single		90%	95%	99%
Batch	Symbol	84.3 $\sigma = 8.12$	88.6 $\sigma = 7.98$	94.0 $\sigma = 4.59$
	Position	79.5 $\sigma = 10.8$	86.6 $\sigma = 7.23$	91.4 $\sigma = 5.37$
Real-time	Symbol	85.6 $\sigma = 7.64$	88.9 $\sigma = 6.84$	93.8 $\sigma = 4.59$
	Position	79.0 $\sigma = 11.5$	84.6 $\sigma = 7.87$	91.6 $\sigma = 4.82$
Multiple		**90%**	**95%**	**99%**
Batch	Symbol	77.3 $\sigma = 16.8$	84.7 $\sigma = 11.8$	94.2 $\sigma = 3.91$
	Position	76.0 $\sigma = 17.1$	80.8 $\sigma = 12.4$	92.6 $\sigma = 4.95$
Real-time	Symbol	80.0 $\sigma = 12.9$	89.1 $\sigma = 7.04$	93.8 $\sigma = 4.38$
	Position	73.2 $\sigma = 13.8$	85.2 $\sigma = 7.72$	91.8 $\sigma = 6.63$

For each subtask, we asked participants to evaluate the recognizer's symbol and position accuracy for the expression set they had just written. The mean accuracies are displayed in Table 3. Participants perceived the recognition accuracy to be no less than 5% below the actual recognition accuracy. Additionally, for lower accuracy levels, participants thought the multiple equation set tasks had lower accuracy than the single equation set tasks.

From the interviews we performed with each participant at the experiment's end, twenty-two participants (91%) reported that they had changed the way they wrote during the experiment. Mostly, people commented that they changed certain aspects of their writing in order to correct perceived errors in the way they wrote (based upon the recognized expressions). Several people reported that they changed the way they wrote super- and subscripts in order to correct those errors; this is not surprising as superscript and subscripts were common in our expressions (see [BL11] for

Table 4: Mean ease in writing expressions. Participants generally found it easy to write expressions regardless of recognition mode.

Set Size	Mode	Accuracy	Mean	σ
Single	Batch	90%	5.875	0.900
		95%	6.208	0.932
		99%	6.542	0.721
	Real-time	90%	6.000	1.216
		95%	6.083	0.929
		99%	6.375	0.711
Multiple	Batch	90%	5.875	1.262
		95%	5.667	1.204
		99%	6.458	0.779
	Real-time	90%	5.375	1.279
		95%	6.125	1.076
		99%	6.292	0.908

Table 5: Mean ease in correcting expressions. As with writing expressions, participants generally found correcting expressions easy regardless of recognition mode.

Set Size	Mode	Accuracy	Mean	σ
Single	Batch	90%	5.708	1.122
		95%	6.167	0.917
		99%	6.417	0.776
	Real-time	90%	5.792	1.141
		95%	5.833	1.090
		99%	6.375	0.875
Multiple	Batch	90%	5.375	1.377
		95%	5.625	1.408
		99%	6.250	1.152
	Real-time	90%	5.708	1.367
		95%	5.833	1.308
		99%	6.375	0.770

Table 6: Mean frustration levels. Lower recognition accuracies lead to higher levels of frustration.

Set Size	Mode	Accuracy	Mean	σ
Single	Batch	90%	3.083	1.349
		95%	2.417	1.742
		99%	1.708	0.859
	Real-time	90%	2.958	1.517
		95%	2.291	1.517
		99%	1.917	0.974
Multiple	Batch	90%	3.375	1.837
		95%	3.042	1.601
		99%	2.333	1.523
	Real-time	90%	3.833	1.685
		95%	2.917	1.530
		99%	1.958	1.042

Table 7: Mean distraction levels. Distraction levels did not exhibit much variance.

Set Size	Mode	Accuracy	Mean	σ
Single	Batch	90%	1.833	1.129
		95%	1.417	0.776
		99%	1.458	0.833
	Real-time	90%	2.083	1.283
		95%	1.875	1.076
		99%	2.250	1.359
Multiple	Batch	90%	1.625	0.875
		95%	1.750	1.113
		99%	1.458	0.833
	Real-time	90%	2.417	1.586
		95%	2.250	1.622
		99%	1.833	1.274

more information on the distribution of errors in the WOZ Math Recognizer). We also asked participants whether they watched the real-time results as they wrote; twenty-one participants reported they had (87%). Of those who did, twelve reported they watched the results, but not all the time during the real-time tasks.

To analyze the data collected for each task, we performed an analysis using Friedman and Wilcoxon Signed Rank tests on the Likert item data [Con98]; we also performed a post-hoc correction using the Holm's Sequential Bonferroni adjustment [Hol79]. For these statements, we compared the data across recognition mode at each accuracy level and set size, across accuracy levels for each recognition mode and set size, and across set size at each recognition mode paired with the two higher accuracy levels (95% and 99%). Average responses can be found in Tables 4 through 6 (recall 1 = strongly disagree, 7 = strongly agree).

3.4.1. Easy To Write

As we expected, there were some significant differences in how easy participants found it to write the expressions at different accuracy levels. The Friedman test for the ease of writing expressions showed significance ($\chi^2_{11} = 63.613$, $p < 0.001$). Increased accuracy made it easier to write the expressions. For the single equation set, increasing accuracy always led to a mean increase in participants' reported ease in writing; the 99% accuracy tasks were easier than the 90% ($Z = -3.358$, $p < 0.0167$) and 95% accuracy tasks ($Z = -2.271$, $p < 0.025$), as were the 95% accuracy tasks compared to the 90% accuracy tasks ($Z = -2.000$, $p < 0.05$). The multiple equation set also had some significant differences in ease of writing based upon accuracy; with the batch recognition mode, the 99% accuracy tasks were easier than the 95% accuracy ($Z = -3.307$, $p < 0.0167$) and 90% accuracy tasks ($Z = -2.274$, $p < 0.025$). Real-time recognition mode produced two significant results; 99% accuracy had greater reported ease in writing than 90% accuracy ($Z = -3.402$, $p < 0.0167$) and 95% accuracy tasks had greater reported ease than 90% accuracy tasks ($Z = -3.080$, $p < 0.025$). Comparing across set size, participants found it easier to write single expressions than multiple expressions using real-time recognition and 90% accuracy ($Z = -2.862$, $p < 0.0167$). No other comparisons for accuracy, nor for set

size nor recognition mode were significant after applying the post-hoc Bonferroni correction.

3.4.2. Easy To Correct

The Friedman test on ease of correction also showed significance ($\chi^2_{11} = 50.656$, $p < 0.001$). As with ease in writing expressions, significant differences in the ease in correcting expressions were found when comparing higher accuracy tasks with lower accuracy tasks. With both the single equation sets and multiple equation sets, participants found it easier to correct higher accuracy tasks than lower accuracy tasks. With batch recognition mode and the single equation sets, participants found it easier to correct expressions at 99% accuracy than at 95% accuracy ($Z = -2.561$, $p < 0.0167$). Using real-time recognition and the single equation sets, participants reported greater ease in correcting expressions at 99% accuracy than at 95% accuracy ($Z = -2.967$, $p < 0.0167$) and at 90% accuracy ($Z = -2.240$, $p < 0.025$). When working with the multiple equation sets and batch recognition mode, 99% recognition accuracy made it easier to correct expressions than 90% accuracy ($Z = -2.662$, $p < 0.0167$). The final significant differences in ease of correction were found with the multiple equation sets and real-time recognition mode; participants found increased ease in correction with 99% accuracy than with 90% ($Z = -2.818$, $p < 0.0167$) and 95% accuracy ($Z = -2.303$, $p < 0.025$). No other comparisons were significant after the Bonferroni correction.

3.4.3. Frustration

For participant reported frustration levels, the Friedman test showed significance ($\chi^2_{11} = 88.66$, $p < 0.0001$). Comparing frustration levels across different accuracies again produced significant differences. Increasing accuracy decreased frustration. When writing in batch recognition mode and the single equation sets, 99% accuracy was less frustrating than 90% accuracy ($Z = -4.122$, $p < 0.0167$) and 95% accuracy ($Z = -2.358$, $p < 0.025$); 95% accuracy was also less frustrating than 90% accuracy ($Z = -2.263$, $p < 0.05$). One significant difference in frustration levels was found when participants used the single equation set and real-time recognition, 99% accuracy was less frustrating than 90% accuracy ($Z = -3.102$, $p < 0.0167$). There were also significant differences across accuracy levels when participants used the multiple equation sets. When working with batch mode, participants reported less frustration when 99% accuracy was used than when 90% ($Z = -3.206$, $p < 0.0167$) or 95% accuracy was used ($Z = -2.428$, $p < 0.025$). Working with real-time recognition, participants reported being less frustrated at 99% accuracy than at 90% accuracy ($Z = -3.868$, $p < 0.0167$) and at 95% accuracy ($Z = -3.216$, $p < 0.025$). They also reported less frustration at 95% accuracy than at 90% accuracy ($Z = -2.829$, $p < 0.05$). No other comparisons across accuracies were significant after post-hoc correction.

Comparing frustration levels across set sizes produced a single significant result. Writing expression groups was more frustrating than writing a single equation; this proved significant with real-time recognition and 90% ($Z = -2.904$, $p < 0.0167$). All other comparisons were not found to be significant.

3.4.4. Distraction

We found significant differences in distraction levels using the Friedman test ($\chi^2_{11} = 32.74$, $p < 0.001$). Comparing across recognition mode, for the single equation set at 95% accuracy, participants found batch recognition less distracting than real-time recognition ($Z = -2.326$, $p < 0.025$). For the multiple equation set at 90% accuracy, participants reported being less distracted using batch recognition than using real-time recognition ($Z = -2.809$, $p < 0.0167$). Comparing across expression set size showed one significant result; when using real-time recognition at 90% accuracy, participants reported being less distracted using the single equation set than with the multiple equation set ($Z = -2.309$, $p = 0.0167$). Comparing distraction levels across accuracies did not reveal any significant results.

4. Discussion

Contrary to our hypothesis, it is clear that recognition accuracy had little impact on user preference for recognition mode, as preference did not generally vary with accuracy. In other words, at different accuracies, participants did not prefer batch or recognition more. Instead, the expression set's size influenced recognition mode preference. As mentioned earlier, we hypothesized that at low accuracies, participants would prefer batch recognition, and as recognition accuracy increased, participants would increasingly prefer real-time recognition over batch recognition. Our experiment's results do not support this hypothesis. We think that participants preferred real-time recognition for the multiple equation set tasks and the 90% accuracy single equation set task, because there were more errors to correct and real-time recognition provided immediate feedback on errors, allowing participants to immediately and easily find and correct recognition errors. As accuracy increased in the single equation set tasks, finding all the errors became easier since there were fewer to find.

In contrast to participant preference for recognition mode, as can be seen in Tables 4 through 6, study participants found it easier to write and correct and were less frustrated at higher recognition accuracies. Interestingly, distraction levels presented an anomaly; in two cases, the mean distraction levels increase from 95% accuracy to 99% accuracy. Only when participants used real-time recognition and the multiple equation set did we see a downward trend in distraction across all three accuracy levels as we expected. In the case of real-time recognition with the single equation set, participants reported greater distraction levels at 99% accuracy than at 90% and 95% accuracy. This trend runs con-

trary to our expectations that increased accuracy would lead to decreased distraction, as fewer errors would distract participants from copying the expressions.

One other anomaly presented itself in the Likert scale responses. When writing the multiple equation set and using batch recognition mode, the mean reported ease in writing the expression decreased from 90% recognition to 95% recognition (but went back up for 99% recognition accuracy). Perhaps this is attributable to some aspect of the expression sets, such as a larger number of exponents or subscripts. It is unclear the exact nature of these anomalies; we will have to experiment further to determine the cause.

Often, participants expressed that batch recognition was better for single expressions, but that real-time was preferable for multiple expressions. One real-time recognition aspect that participants liked was its immediacy; it gave them immediate feedback and they were able to immediately correct and adapt their writing styles. For tasks at 90% recognition accuracy, some participants felt that batch mode was tedious, as there were many errors and it was hard to remember which errors they had corrected before hitting the recognize button. This was especially true with the multiple equation set, since there were more errors to correct, which forced participants to spend a long period performing error correction. Additionally, participants stated that correcting all their mistakes at once was time consuming. Unfortunately, we did not time how long it took each participant to write and correct the expressions, so we cannot verify or refute this perception. This brings us back to an inherent issue with batch recognition; it requires a period of intense visual identification of errors. We think that this explains why there was a stronger preference for real-time recognition for the low-accuracy multiple equation task compared to the higher accuracy tasks.

Participants often stated that they wanted to spend time writing expressions; interestingly, this was often used as a reason for preferring both batch and real-time recognition. We think that differences in what distracted participants explains this contradiction. Participants who were distracted by recognition errors as they wrote would find it easier and faster to write and then recognize; participants who were not distracted would not have to sit through an error correction cycle of finding and rewriting incorrectly recognized symbols. Since participants reported lower distraction levels using batch recognition over real-time recognition, we can speculate that distraction played little part in user preference for recognition mode. Additionally, there may have been differences in writing cycle perceptions; some participants may have included error correction as part of writing, while others did not. Those who viewed error correction as separate would likely view any period of solely correcting errors as "not writing."

In general, participants expressed a desire for faster recognition and correction; that is, they wanted to see recognized expressions immediately and fix mistakes immediately. During the final interview with the participants, we asked them whether they would prefer a version of batch recognition where they could press the recognize button after each expression or correction, over the batch recognition performed in the study. This alternate batch recognition would give more immediate recognized expressions and the ability for users to see error correction results almost immediately. Thirteen participants expressed that they would like that version of batch, ten participants stated that they preferred the implemented version, and one stated that it was situation dependent.

Participants consistently underestimated the position and symbol recognition accuracy for each task. The perceived accuracies were fairly consistently 5% or greater in error. What is most interesting is that participants had a greater underestimation of accuracies for the multiple equation tasks at low recognition accuracies. Additionally, participants exhibited a greater variation in perceived recognition accuracy for the multiple equation set than the single equation set (and variation decreased as the real accuracy increased). One explanation might be that participants saw several expressions with errors and viewed all the errors as effecting one expression; participants may have also had a harder time evaluating the number of symbols and positions for multiple expressions at once.

Although it is not the primary task that most users will perform while using mathematical handwriting recognition software, we chose to have participants perform a copying task during the study, as we felt that it was a representative task of one type of task that users perform in real-world situations (such as in educational settings). For instance, students will copy down equations while doing their homework and teachers might copy them down while creating a test. Additionally, using a copying task allowed us to control the experiment and what participants wrote.

One thing to note is the real-time recognition mode in the WOZ Math Recognizer sometimes occasionally suffers from delays due to the wizard having to determine the most appropriate recognition feedback. However, these delays are minimal and do not necessarily disrupt the flow of the recognized mathematics. We consider this real-time feedback to be very close to what you would get with a real math expression recognizer. The results from the experiment did not show that this was an issue.

We feel that knowing user perceptions about the two recognition modes are more important for user interface design than an objective quantitative measurement of the modes, such as user speed. Consequently, a subjective quantitative experiment was performed; objectively measuring those perceptions would have greatly increased the setup's complexity and the time required of each participant.

5. Future Work

As discussed earlier, participants engaged in a copying task, which is not the most common mathematics task, but still an important one. A thinking task where the participants are doing something with the mathematics is an important area that we need to explore, especially in intelligent tutoring applications; people's preferences may change when the task changes. One way in which we might simulate a thinking task is by combining a copying task and a distraction task.

6. Conclusion

In this paper, we explored how recognition accuracy and the number of expressions written affect user preference for recognition mode and their perceptions of the recognizer. Contrary to our expectations, recognition accuracy had little effect on user preference for recognition mode. In fact, the number of expressions that a user wrote at a time had the most effect on preference for mode; with more than one expression, users preferred real-time recognition over batch recognition. At high accuracies with single expressions, there was no real preference for recognition now or later.

While recognition accuracy is important for a good user experience, it is not the holy grail of recognizer properties. More accurate recognizers are easier to use, but an accurate recognizer can still be distracting to users.

When users copy down large expressions or many expressions at a time, they generally want to see immediate recognition feedback. Participants consistently preferred real-time recognition over batch recognition when they wrote multiple expressions together.

Preference for recognition mode is a personal choice; some people look at recognition feedback while they write, while others don't. Our results tell user interface designers that the choice of recognition mode is better left up to the user. While participants preferred real-time recognition mode overall (105 to 39 tasks), it was not overwhelmingly preferred for all factors and levels. As well, participants sometimes had more positive feelings about batch recognition than real-time recognition. By allowing the user to control whether they use real-time or batch recognition, designers can provide a mathematics handwriting recognizer that is easy to use, and minimally distracting and frustrating.

7. Acknowledgements

This work is supported in part by NSF CAREER award IIS-0845921 and NSF awards IIS-0856045 and CCF-1012056.

References

[Ant08] ANTHONY L.: *Developing Handwriting-based Intelligent Tutors to Enhance Mathematics Learning*. PhD thesis, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2008. 1, 2

[BL11] BOTT J. N., LAVIOLA JR. J. J.: *The WOZ Math Recognizer: A Mathematics Handwriting Recognition Wizard of Oz Tool*. Tech. Rep. CS-TR-11-03, University of Central Florida, 2011. 1, 2, 4

[Con98] CONOVER W. J.: *Practical Nonparametric Statistics*. John Wiley & Sons, Dec. 1998. 5

[CY01] CHAN K.-F., YEUNG D.-Y.: Error detection, error correction and performance evaluation in on-line mathematical expression recognition. *Pattern Recognition 34*, 8 (2001), 1671 – 1684. 1

[FGP*09] FIELD M., GORDON S., PETERSON E., ROBINSON R., STAHOVICH T., ALVARADO C.: The effect of task on classification accuracy: using gesture recognition techniques in free-sketch recognition. In *Proceedings of the 6th Eurographics Symposium on Sketch-Based Interfaces and Modeling* (New York, NY, USA, 2009), SBIM '09, ACM, pp. 109–116. 2

[Hol79] HOLM S.: A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics 6*, 2 (1979), pp. 65–70. 5

[LaL94] LALOMIA M.: User acceptance of handwritten recognition accuracy. In *Conference companion on Human factors in computing systems* (New York, NY, USA, 1994), CHI '94, ACM, pp. 107–108. 2

[LaV07] LAVIOLA JR. J. J.: An initial evaluation of mathpad^2: A tool for creating dynamic mathematical illustrations. *Computers & Graphics 31* (August 2007), 540–553. 1

[LLMZ08] LAVIOLA JR. J. J., LEAL A., MILLER T. S., ZELEZNIK R. C.: Evaluation of techniques for visualizing mathematical expression recognition results. In *Proceedings of graphics interface 2008* (Toronto, Ont., Canada, 2008), GI '08, Canadian Information Processing Society, pp. 131–138. 2

[Mic09] MICROSOFT: Math input panel. Computer program, 2009. 3

[NP95] NOUBOUD F., PLAMONDON R.: Document image analysis. IEEE Computer Society Press, Los Alamitos, CA, USA, 1995, ch. On-line recognition of handprinted characters: survey and beta tests, pp. 342–355. 1

[OSA08] OVIATT S., SWINDELLS C., ARTHUR A.: Implicit user-adaptive system engagement in speech and pen interfaces. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems* (New York, NY, USA, 2008), CHI '08, ACM, pp. 969–978. 2

[RMC03] READ J. C., MACFARLANE S., CASEY C.: 'good enough for what?': acceptance of handwriting recognition errors by child users. In *Proceedings of the 2003 conference on Interaction design and children* (New York, NY, USA, 2003), IDC '03, ACM, pp. 155–155. 2

[SNA99] SMITHIES S., NOVINS K., ARVO J.: A handwriting-based equation editor. In *Proceedings of the 1999 conference on Graphics interface '99* (San Francisco, CA, USA, 1999), Morgan Kaufmann Publishers Inc., pp. 84–91. 1

[WWA07] WAIS P., WOLIN A., ALVARADO C.: Designing a sketch recognition front-end: user perception of interface elements. In *Proceedings of the 4th Eurographics workshop on Sketch-based interfaces and modeling* (New York, NY, USA, 2007), SBIM '07, ACM, pp. 99–106. 2

[ZMLL08] ZELEZNIK R., MILLER T., LI C., LAVIOLA JR. J. J.: Mathpaper: Mathematical sketching with fluid support for interactive computation. In *SG '08: Proceedings of the 9th international symposium on Smart Graphics* (Berlin, Heidelberg, 2008), Springer-Verlag, pp. 20–32. 3